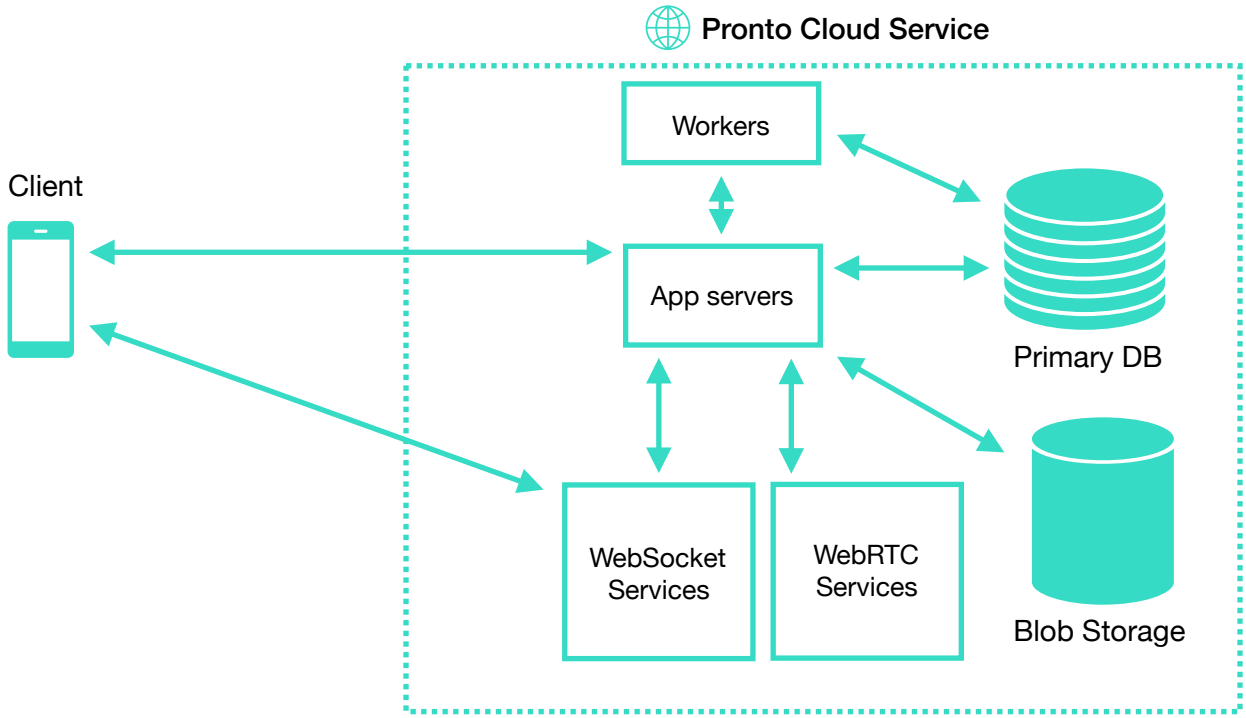
Author: Ben Dolman
CTO, Hit Labs Inc.
Updated 3/22/2020

# Pronto: Technical Architecture

Pronto is a real-time communication tool built on top of modern technology such as WebSockets and WebRTC.

This diagram illustrates a high-level technical architecture. Please see the separate Pronto Network Diagram for more detail. The Pronto Cloud Service is primarily hosted in AWS in the US-East-1 region.




## App Servers

App servers are the front-line workhorses of the Pronto Cloud Service, handling HTTP requests from Pronto clients as well as integrated services. These servers are load balanced and, because they are stateless, horizontally scalable to whatever amount is needed. Demand is managed via auto-scaling rules that will automatically add or remove servers as needed based on load.

Code for app servers is written in PHP using the Laravel framework. JSON is used as the transport format.

## Workers

Asynchronous, scheduled, repetitive, or long-running tasks are pushed to a queue of worker servers. Examples of these jobs include video transcoding, language translation, and push notifications. Each of these servers monitor an SQS queue for new jobs and then signal the success or failure of those jobs. Each worker is stateless which allows for horizontal scalability — we can add as many new worker servers as we want to satisfy demand. The number of workers in the pool is scaled automatically based on a number of factors including CPU usage and queue size.

## Provisioning Services

Auto-provisioning of users and groups is available for several third-party systems, including Canvas by Instructure and Blackboard. Hit Labs runs integration servers that sync with these third-party systems at a regular interval and determine if users and groups needed to be added or removed from synced organizations.

## Data Storage

Hit Labs utilizes several data stores to provide the Pronto Cloud Service. The primary datastore is a ClustrixDB database cluster. ClustrixDB is a distributed database with a MySQL interface that allows for unlimited horizontal scaling for both reads and writes while still providing full ACID compliance.

Pronto uses a multi-tenant architecture with a cluster of ClustrixDB nodes that can be expanded as needed. We currently run eight nodes, each of which uses full disk encryption.

Additional data storage includes RDS MySQL instances for reporting, and Redis for high performance app state caching.

S3 is used for blob storage (photos, videos, files). All S3 data is encrypted.

## Backup

Our distributed database is highly fault tolerant, maintaining at least 2 replicas of each piece of data on different hardware nodes, allowing for multiple simultaneous hardware failures. In addition, data is backed up every 2 hours to separate persistent encrypted storage (S3) across multiple regions allowing for minimal-loss recovery in the event of an unlikely catastrophic failure of our entire primary database cluster.

S3 is used for blob storage (photos, videos, files) and provides industry-leading durability and fault-tolerance.

## Clients

Pronto has Web, Android and iOS clients. All three are written as native applications to account for platform differences and optimize user interface performance, which is critical for a real-time messaging platform.

**Web (Chrome, Firefox, Safari)**: PHP, Laravel, VueJS
**iOS (9.2+)**: Swift, Objective-C, Core Data
**Android (4.4+)**: Kotlin, Java, Realm

## Third-Party Services

Like most cloud services, Hit Labs uses outside vendors to provide building blocks and power portions of its services.

| Feature | Vendors | Notes |
|---|---|---|
| **Real-Time Messaging** | Pusher | Clients open WebSocket connections via Pusher. Model updates (such as new messages, user changes, etc.) are pushed down these connections. |
| **Livestream Video** | TokBox | TokBox provides a set of WebRTC primitives and a cloud service that we use to drive a custom live-streaming interface available across our platforms. |
| **Photo, Video, File Uploads** | AWS S3 | Files are uploaded to S3 and then processed in a worker job. |
| **Push Notifications** | AWS SNS | AWS SNS is used as a bridge to simplify our interface to Apple APNs and Google GCM for mobile push notifications. |
| **Translation** | Google Translate | Pronto can auto-detect the language of a message and machine-translate it. The content of each message is processed through Google Translate via a worker job. |
| **Link Previews** | embed.ly | When you message someone a URL, we generate a preview using a third-party service. |
| **Video Transcoding** | AWS Elastic Transcoding | Pronto transcodes large, user-uploaded videos to sizes more easily consumable on mobile devices. |
| **GIFs** | Giphy | Users can search and select GIFs to share in their groups. These search results and images are handled by Giphy. |
| **Phone Verification** | Twilio | Phone verification requests are sent via Twilio. Only applies to users that login using a phone number instead of email. |
| **Email Verification** | AWS SMS | Email verification requests are sent via AWS SMS. |
| **Server Logging and Analytics** | NewRelic, Loggly | We use several third-party providers for server and log monitoring. |

| Feature | Vendors | Notes |
| --- | --- | --- |
| **Client-side Analytics** | Apple, Google, Fabric | We report usage statistics in our mobile products to a third-party service in order to track active user count, crash reports, bug reports, etc. |